**Week - 5**

**Microservices**

**Mandatory Hands-on**

**Creating Microservices for account and loan**

1. For Account

package com.cognizant.account;

import org.springframework.web.bind.annotation.\*;

*@RestController*

*@RequestMapping*("/accounts")

public class AccountController {

*@GetMapping*("/{number}")

public Account getAccountByNumber(*@PathVariable* String number) {

return new Account(number, "Savings", 927642);

}

static class Account {

private String number;

private String type;

private double balance;

public Account(String number, String type, double balance) {

this.number = number;

this.type = type;

this.balance = balance;

}

public String getNumber() { return number; }

public void setNumber(String number) { this.number = number; }

public String getType() { return type; }

public void setType(String type) { this.type = type; }

public double getBalance() { return balance; }

public void setBalance(double balance) { this.balance = balance; }

}

}

package com.cognizant.account;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class AccountApplication {

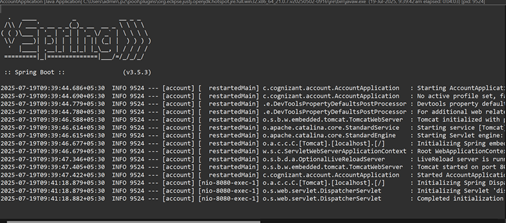
public static void main(String[] args) {

SpringApplication.*run*(AccountApplication.class, args);

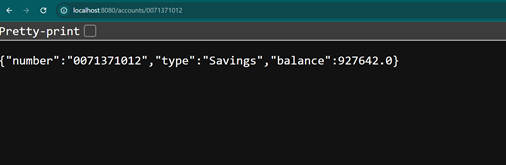
}

}

Console:



Output:



1. For loan

package com.cognizant.loan;

import org.springframework.web.bind.annotation.\*;

*@RestController*

*@RequestMapping*("/loans")

public class LoanController {

*@GetMapping*("/{loanNumber}")

public Loan getLoanDetails(*@PathVariable* String loanNumber) {

return new Loan(loanNumber, "Personal", 280000, 7.5);

}

static class Loan {

private String loanNumber;

private String type;

private double amount;

private double interestRate;

public Loan(String loanNumber, String type, double amount, double interestRate) {

this.loanNumber = loanNumber;

this.type = type;

this.amount = amount;

this.interestRate = interestRate;

}

public String getLoanNumber() { return loanNumber; }

public void setLoanNumber(String loanNumber) { this.loanNumber = loanNumber; }

public String getType() { return type; }

public void setType(String type) { this.type = type; }

public double getAmount() { return amount; }

public void setAmount(double amount) { this.amount = amount; }

public double getInterestRate() { return interestRate; }

public void setInterestRate(double interestRate) { this.interestRate = interestRate; }

}

}

package com.cognizant.loan;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class LoanApplication {

public static void main(String[] args) {

SpringApplication.*run*(LoanApplication.class, args);

}

}

. \_\_\_\_ \_ \_\_ \_ \_

/\\ / \_\_\_'\_ \_\_ \_ \_(\_)\_ \_\_ \_\_ \_ \ \ \ \

( ( )\\_\_\_ | '\_ | '\_| | '\_ \/ \_` | \ \ \ \

\\/ \_\_\_)| |\_)| | | | | || (\_| | ) ) ) )

' |\_\_\_\_| .\_\_|\_| |\_|\_| |\_\\_\_, | / / / /

=========|\_|==============|\_\_\_/=/\_/\_/\_/

:: Spring Boot :: (v3.5.3)

2025-07-19T12:55:27.429+05:30 INFO 6964 --- [loan] [ restartedMain] com.cognizant.loan.LoanApplication : Starting LoanApplication using Java 21.0.7 with PID 6964 (D:\7282\microservices\loan\target\classes started by admin in D:\7282\microservices\loan)

2025-07-19T12:55:27.435+05:30 INFO 6964 --- [loan] [ restartedMain] com.cognizant.loan.LoanApplication : No active profile set, falling back to 1 default profile: "default"

2025-07-19T12:55:27.541+05:30 INFO 6964 --- [loan] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools property defaults active! Set 'spring.devtools.add-properties' to 'false' to disable

2025-07-19T12:55:27.542+05:30 INFO 6964 --- [loan] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additional web related logging consider setting the 'logging.level.web' property to 'DEBUG'

2025-07-19T12:55:29.172+05:30 INFO 6964 --- [loan] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8080 (http)

2025-07-19T12:55:29.196+05:30 INFO 6964 --- [loan] [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]

2025-07-19T12:55:29.197+05:30 INFO 6964 --- [loan] [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]

2025-07-19T12:55:29.247+05:30 INFO 6964 --- [loan] [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext

2025-07-19T12:55:29.250+05:30 INFO 6964 --- [loan] [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1703 ms

2025-07-19T12:55:29.781+05:30 INFO 6964 --- [loan] [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running on port 35729

2025-07-19T12:55:29.834+05:30 INFO 6964 --- [loan] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080 (http) with context path '/'

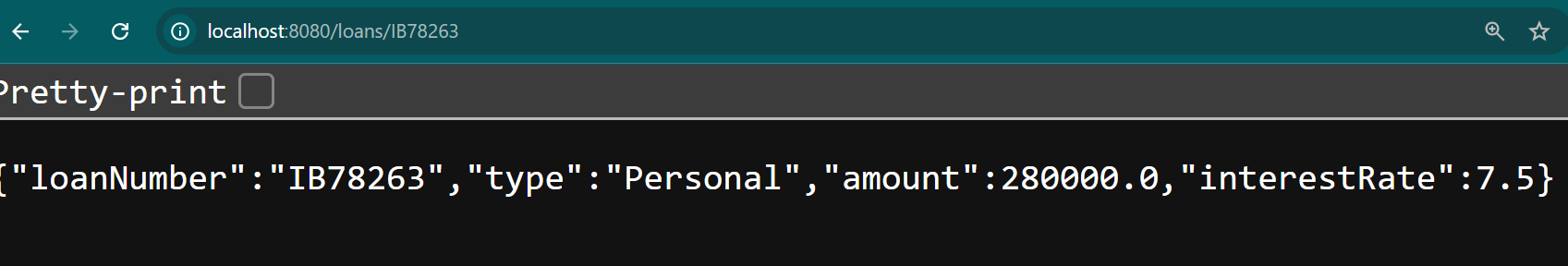
2025-07-19T12:55:29.847+05:30 INFO 6964 --- [loan] [ restartedMain] com.cognizant.loan.LoanApplication : Started LoanApplication in 3.082 seconds (process running for 3.789)

2025-07-19T12:56:02.053+05:30 INFO 6964 --- [loan] [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'

2025-07-19T12:56:02.053+05:30 INFO 6964 --- [loan] [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'

2025-07-19T12:56:02.055+05:30 INFO 6964 --- [loan] [nio-8080-exec-1] o.s.web.servlet.DispatcherServlet : Completed initialization in 2 ms

Output:



**Additional Hands-on**

**Create Eureka Discovery Server and register microservices**

package com.cognizant.eureka\_discovery\_server;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.cloud.netflix.eureka.server.EnableEurekaServer;

*@EnableEurekaServer*

*@SpringBootApplication*

public class EurekaDiscoveryServerApplication {

public static void main(String[] args) {

SpringApplication.*run*(EurekaDiscoveryServerApplication.class, args);

}

}

spring.application.name=eureka-discovery-server

server.port=8761

eureka.client.register-with-eureka=false

eureka.client.fetch-registry=false

logging.level.com.netflix.eureka=OFF

logging.level.com.netflix.discovery=OFF

**Console:**

. \_\_\_\_ \_ \_\_ \_ \_

/\\ / \_\_\_'\_ \_\_ \_ \_(\_)\_ \_\_ \_\_ \_ \ \ \ \

( ( )\\_\_\_ | '\_ | '\_| | '\_ \/ \_` | \ \ \ \

\\/ \_\_\_)| |\_)| | | | | || (\_| | ) ) ) )

' |\_\_\_\_| .\_\_|\_| |\_|\_| |\_\\_\_, | / / / /

=========|\_|==============|\_\_\_/=/\_/\_/\_/

:: Spring Boot :: (v3.5.3)

2025-07-19T17:32:20.808+05:30 INFO 28820 --- [eureka-discovery-server] [ main] c.c.e.EurekaDiscoveryServerApplication : Starting EurekaDiscoveryServerApplication using Java 21.0.7 with PID 28820 (D:\7282\microservices\eureka-discovery-server\target\classes started by admin in D:\7282\microservices\eureka-discovery-server)

2025-07-19T17:32:20.817+05:30 INFO 28820 --- [eureka-discovery-server] [ main] c.c.e.EurekaDiscoveryServerApplication : No active profile set, falling back to 1 default profile: "default"

2025-07-19T17:32:23.198+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.s.cloud.context.scope.GenericScope : BeanFactory id=d4b5e39c-8d2d-3834-94f8-dc636db86bcb

2025-07-19T17:32:23.894+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8761 (http)

2025-07-19T17:32:23.924+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.apache.catalina.core.StandardService : Starting service [Tomcat]

2025-07-19T17:32:23.924+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]

2025-07-19T17:32:24.055+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext

2025-07-19T17:32:24.057+05:30 INFO 28820 --- [eureka-discovery-server] [ main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 3097 ms

2025-07-19T17:32:26.572+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.s.v.b.OptionalValidatorFactoryBean : Failed to set up a Bean Validation provider: jakarta.validation.NoProviderFoundException: Unable to create a Configuration, because no Jakarta Bean Validation provider could be found. Add a provider like Hibernate Validator (RI) to your classpath.

2025-07-19T17:32:28.015+05:30 WARN 28820 --- [eureka-discovery-server] [ main] iguration$LoadBalancerCaffeineWarnLogger : Spring Cloud LoadBalancer is currently working with the default cache. While this cache implementation is useful for development and tests, it's recommended to use Caffeine cache in production.You can switch to using Caffeine cache, by adding it and org.springframework.cache.caffeine.CaffeineCacheManager to the classpath.

2025-07-19T17:32:28.064+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.s.c.n.eureka.InstanceInfoFactory : Setting initial instance status as: STARTING

2025-07-19T17:32:28.594+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.s.b.a.e.web.EndpointLinksResolver : Exposing 1 endpoint beneath base path '/actuator'

2025-07-19T17:32:28.744+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.s.c.n.e.s.EurekaServiceRegistry : Registering application EUREKA-DISCOVERY-SERVER with eureka with status UP

2025-07-19T17:32:28.770+05:30 INFO 28820 --- [eureka-discovery-server] [ Thread-9] o.s.c.n.e.server.EurekaServerBootstrap : isAws returned false

2025-07-19T17:32:28.771+05:30 INFO 28820 --- [eureka-discovery-server] [ Thread-9] o.s.c.n.e.server.EurekaServerBootstrap : Initialized server context

2025-07-19T17:32:28.783+05:30 INFO 28820 --- [eureka-discovery-server] [ Thread-9] e.s.EurekaServerInitializerConfiguration : Started Eureka Server

2025-07-19T17:32:28.818+05:30 INFO 28820 --- [eureka-discovery-server] [ main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8761 (http) with context path '/'

2025-07-19T17:32:28.820+05:30 INFO 28820 --- [eureka-discovery-server] [ main] .s.c.n.e.s.EurekaAutoServiceRegistration : Updating port to 8761

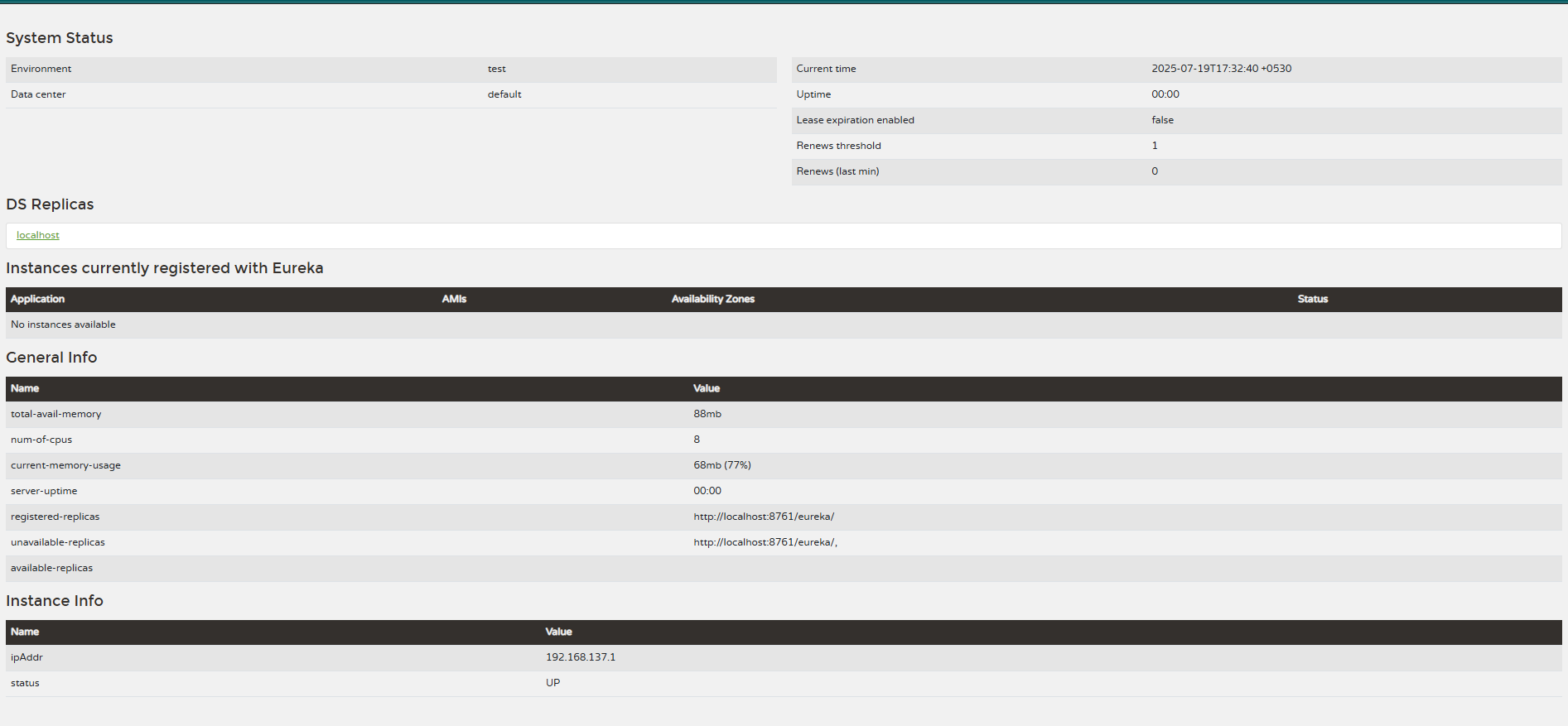
2025-07-19T17:32:28.879+05:30 INFO 28820 --- [eureka-discovery-server] [ main] c.c.e.EurekaDiscoveryServerApplication : Started EurekaDiscoveryServerApplication in 9.161 seconds (process running for 10.53)

2025-07-19T17:32:40.788+05:30 INFO 28820 --- [eureka-discovery-server] [nio-8761-exec-6] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring DispatcherServlet 'dispatcherServlet'

2025-07-19T17:32:40.788+05:30 INFO 28820 --- [eureka-discovery-server] [nio-8761-exec-6] o.s.web.servlet.DispatcherServlet : Initializing Servlet 'dispatcherServlet'

2025-07-19T17:32:40.790+05:30 INFO 28820 --- [eureka-discovery-server] [nio-8761-exec-6] o.s.web.servlet.DispatcherServlet : Completed initialization in 2 ms

**Localhost:8761**



1. For account

package com.cognizant.account;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class AccountApplication {

public static void main(String[] args) {

SpringApplication.*run*(AccountApplication.class, args);

}

}

package com.cognizant.account;

import org.springframework.web.bind.annotation.\*;

import java.util.\*;

*@RestController*

*@RequestMapping*("/accounts")

public class AccountController {

*@GetMapping*("/{number}")

public Map<String,Object> getAccount(*@PathVariable* String number){

return Map.*of*(

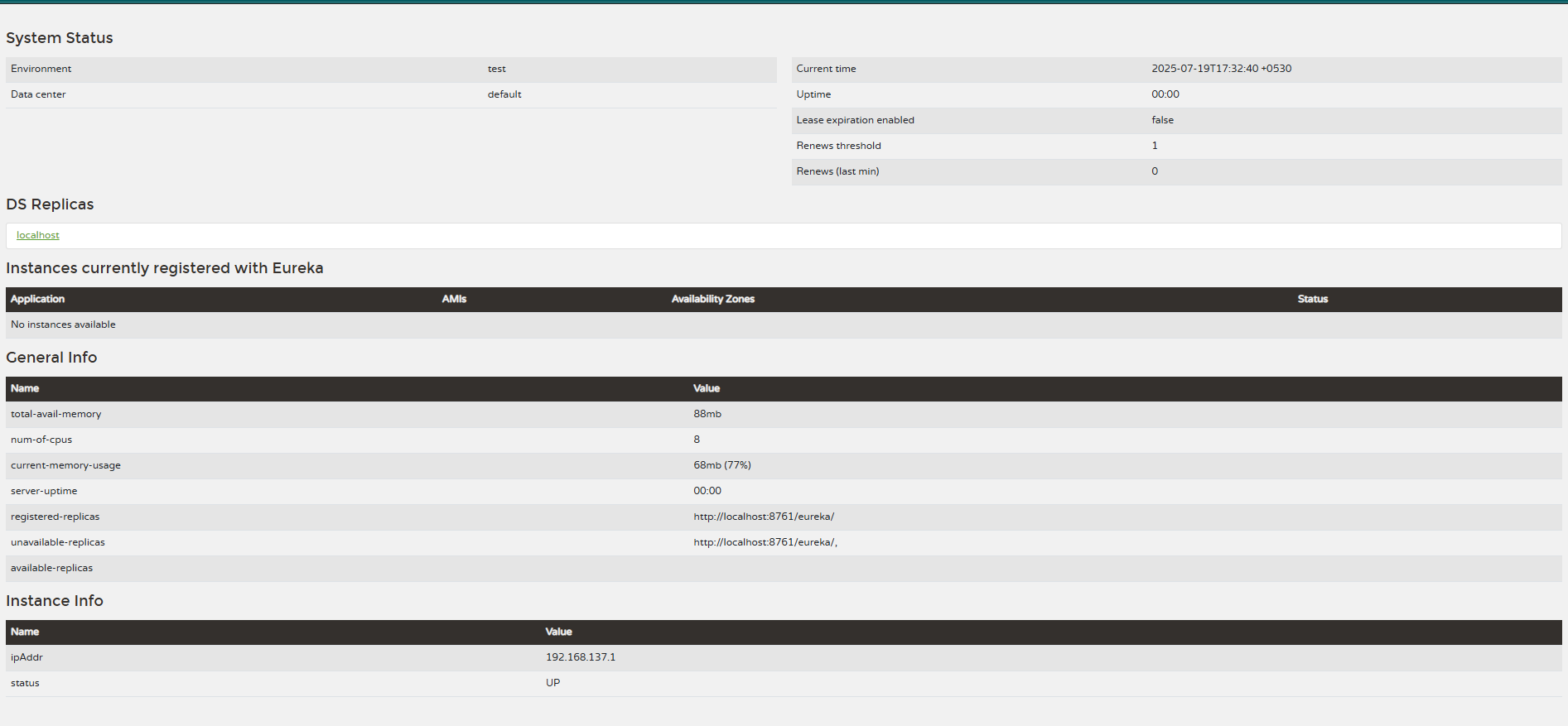
"number", number,

"type", "savings","balance" , 28173);

}

}

Upon running eureka-discovery-server



Secondly, upon running account

. \_\_\_\_ \_ \_\_ \_ \_

/\\ / \_\_\_'\_ \_\_ \_ \_(\_)\_ \_\_ \_\_ \_ \ \ \ \

( ( )\\_\_\_ | '\_ | '\_| | '\_ \/ \_` | \ \ \ \

\\/ \_\_\_)| |\_)| | | | | || (\_| | ) ) ) )

' |\_\_\_\_| .\_\_|\_| |\_|\_| |\_\\_\_, | / / / /

=========|\_|==============|\_\_\_/=/\_/\_/\_/

:: Spring Boot :: (v3.5.3)

2025-07-20T19:51:34.803+05:30 INFO 31996 --- [account] [ restartedMain] c.cognizant.account.AccountApplication : Starting AccountApplication using Java 21.0.7 with PID 31996 (D:\7282\microservices\account\target\classes started by admin in D:\7282\microservices\account)

2025-07-20T19:51:34.806+05:30 INFO 31996 --- [account] [ restartedMain] c.cognizant.account.AccountApplication : No active profile set, falling back to 1 default profile: "default"

2025-07-20T19:51:34.914+05:30 INFO 31996 --- [account] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools property defaults active! Set 'spring.devtools.add-properties' to 'false' to disable

2025-07-20T19:51:34.915+05:30 INFO 31996 --- [account] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additional web related logging consider setting the 'logging.level.web' property to 'DEBUG'

2025-07-20T19:51:36.634+05:30 INFO 31996 --- [account] [ restartedMain] o.s.cloud.context.scope.GenericScope : BeanFactory id=f2cea789-6c08-3c31-b460-448982ab026a

2025-07-20T19:51:37.280+05:30 INFO 31996 --- [account] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8080 (http)

2025-07-20T19:51:37.299+05:30 INFO 31996 --- [account] [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]

2025-07-20T19:51:37.300+05:30 INFO 31996 --- [account] [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]

2025-07-20T19:51:37.374+05:30 INFO 31996 --- [account] [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext

2025-07-20T19:51:37.376+05:30 INFO 31996 --- [account] [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 2459 ms

2025-07-20T19:51:38.442+05:30 INFO 31996 --- [account] [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running on port 35729

2025-07-20T19:51:38.578+05:30 INFO 31996 --- [account] [ restartedMain] DiscoveryClientOptionalArgsConfiguration : Eureka HTTP Client uses RestTemplate.

2025-07-20T19:51:38.677+05:30 WARN 31996 --- [account] [ restartedMain] iguration$LoadBalancerCaffeineWarnLogger : Spring Cloud LoadBalancer is currently working with the default cache. While this cache implementation is useful for development and tests, it's recommended to use Caffeine cache in production.You can switch to using Caffeine cache, by adding it and org.springframework.cache.caffeine.CaffeineCacheManager to the classpath.

2025-07-20T19:51:38.726+05:30 INFO 31996 --- [account] [ restartedMain] o.s.c.n.eureka.InstanceInfoFactory : Setting initial instance status as: STARTING

2025-07-20T19:51:38.783+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Initializing Eureka in region us-east-1

2025-07-20T19:51:38.800+05:30 INFO 31996 --- [account] [ restartedMain] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints via configuration

2025-07-20T19:51:38.825+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Disable delta property : false

2025-07-20T19:51:38.825+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Single vip registry refresh property : null

2025-07-20T19:51:38.825+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Force full registry fetch : false

2025-07-20T19:51:38.825+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Application is null : false

2025-07-20T19:51:38.826+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Registered Applications size is zero : true

2025-07-20T19:51:38.826+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Application version is -1: true

2025-07-20T19:51:38.826+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Getting all instance registry info from the eureka server

2025-07-20T19:51:39.724+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : The response status is 200

2025-07-20T19:51:39.726+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Starting heartbeat executor: renew interval is: 30

2025-07-20T19:51:39.728+05:30 INFO 31996 --- [account] [ restartedMain] c.n.discovery.InstanceInfoReplicator : InstanceInfoReplicator onDemand update allowed rate per min is 4

2025-07-20T19:51:39.733+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Discovery Client initialized at timestamp 1753021299730 with initial instances count: 0

2025-07-20T19:51:39.739+05:30 INFO 31996 --- [account] [ restartedMain] o.s.c.n.e.s.EurekaServiceRegistry : Registering application ACCOUNT with eureka with status UP

2025-07-20T19:51:39.740+05:30 INFO 31996 --- [account] [ restartedMain] com.netflix.discovery.DiscoveryClient : Saw local status change event StatusChangeEvent [timestamp=1753021299740, current=UP, previous=STARTING]

2025-07-20T19:51:39.743+05:30 INFO 31996 --- [account] [foReplicator-%d] com.netflix.discovery.DiscoveryClient : DiscoveryClient\_ACCOUNT/DESKTOP-KQJ5G19:account:8080: registering service...

2025-07-20T19:51:39.794+05:30 INFO 31996 --- [account] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080 (http) with context path '/'

2025-07-20T19:51:39.797+05:30 INFO 31996 --- [account] [ restartedMain] .s.c.n.e.s.EurekaAutoServiceRegistration : Updating port to 8080

2025-07-20T19:51:39.822+05:30 INFO 31996 --- [account] [ restartedMain] c.cognizant.account.AccountApplication : Started AccountApplication in 5.851 seconds (process running for 7.029)

2025-07-20T19:51:39.997+05:30 INFO 31996 --- [account] [foReplicator-%d] com.netflix.discovery.DiscoveryClient : DiscoveryClient\_ACCOUNT/DESKTOP-KQJ5G19:account:8080 - registration status: 204

2025-07-20T19:52:09.742+05:30 INFO 31996 --- [account] [reshExecutor-%d] com.netflix.discovery.DiscoveryClient : Disable delta property : false

2025-07-20T19:52:09.743+05:30 INFO 31996 --- [account] [reshExecutor-%d] com.netflix.discovery.DiscoveryClient : Single vip registry refresh property : null

2025-07-20T19:52:09.743+05:30 INFO 31996 --- [account] [reshExecutor-%d] com.netflix.discovery.DiscoveryClient : Force full registry fetch : false

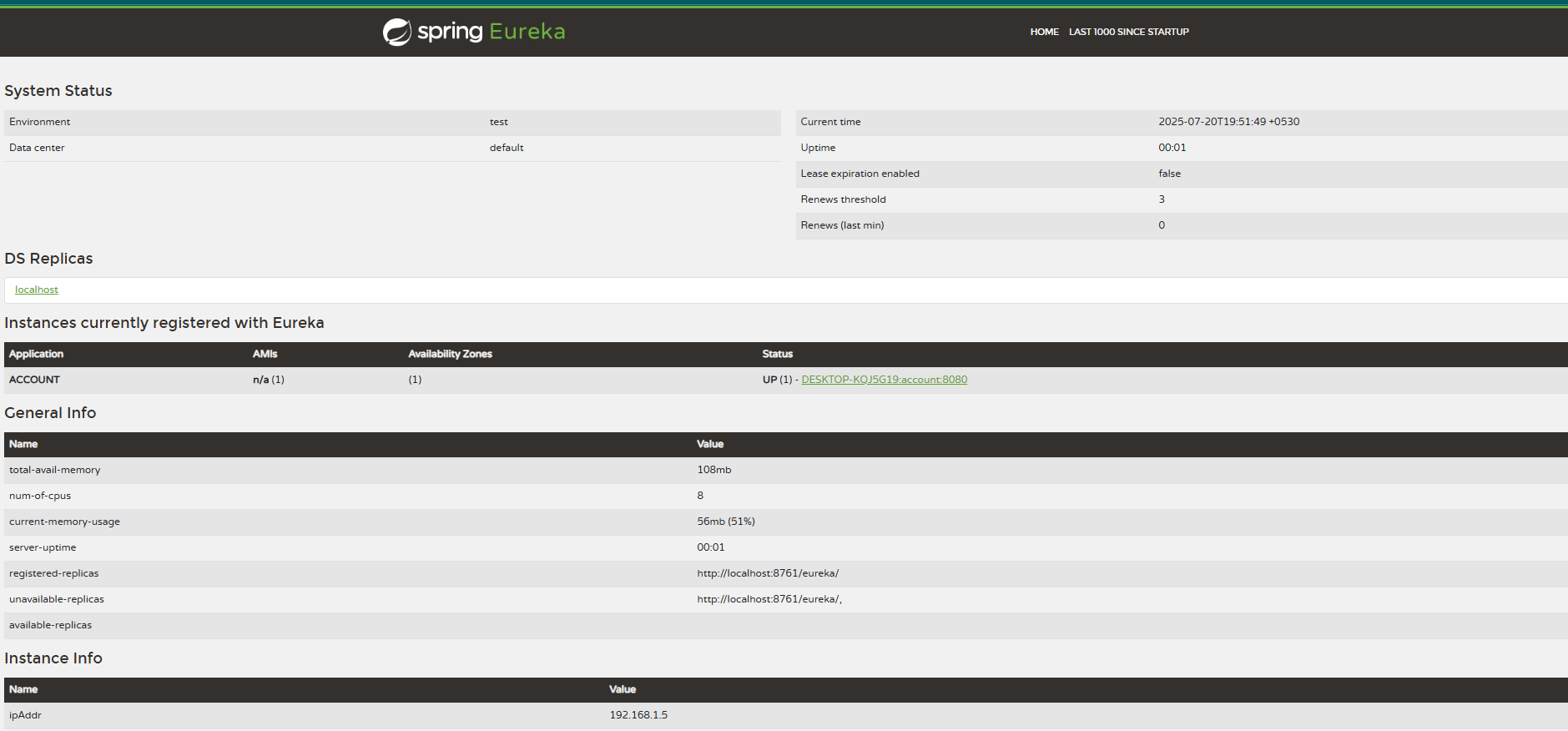
2025-07-20T19:52:09.743+05:30 INFO 31996 --- [account] [reshExecutor-%d] com.netflix.discovery.DiscoveryClient : Application is null : false

2025-07-20T19:52:09.744+05:30 INFO 31996 --- [account] [reshExecutor-%d] com.netflix.discovery.DiscoveryClient : Registered Applications size is zero : true

2025-07-20T19:52:09.744+05:30 INFO 31996 --- [account] [reshExecutor-%d] com.netflix.discovery.DiscoveryClient : Application version is -1: false

2025-07-20T19:52:09.745+05:30 INFO 31996 --- [account] [reshExecutor-%d] com.netflix.discovery.DiscoveryClient : Getting all instance registry info from the eureka server

2025-07-20T19:52:09.856+05:30 INFO 31996 --- [account] [reshExecutor-%d] com.netflix.discovery.DiscoveryClient : The response status is 200



1. For loan

package com.cognizant.loan;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class LoanApplication {

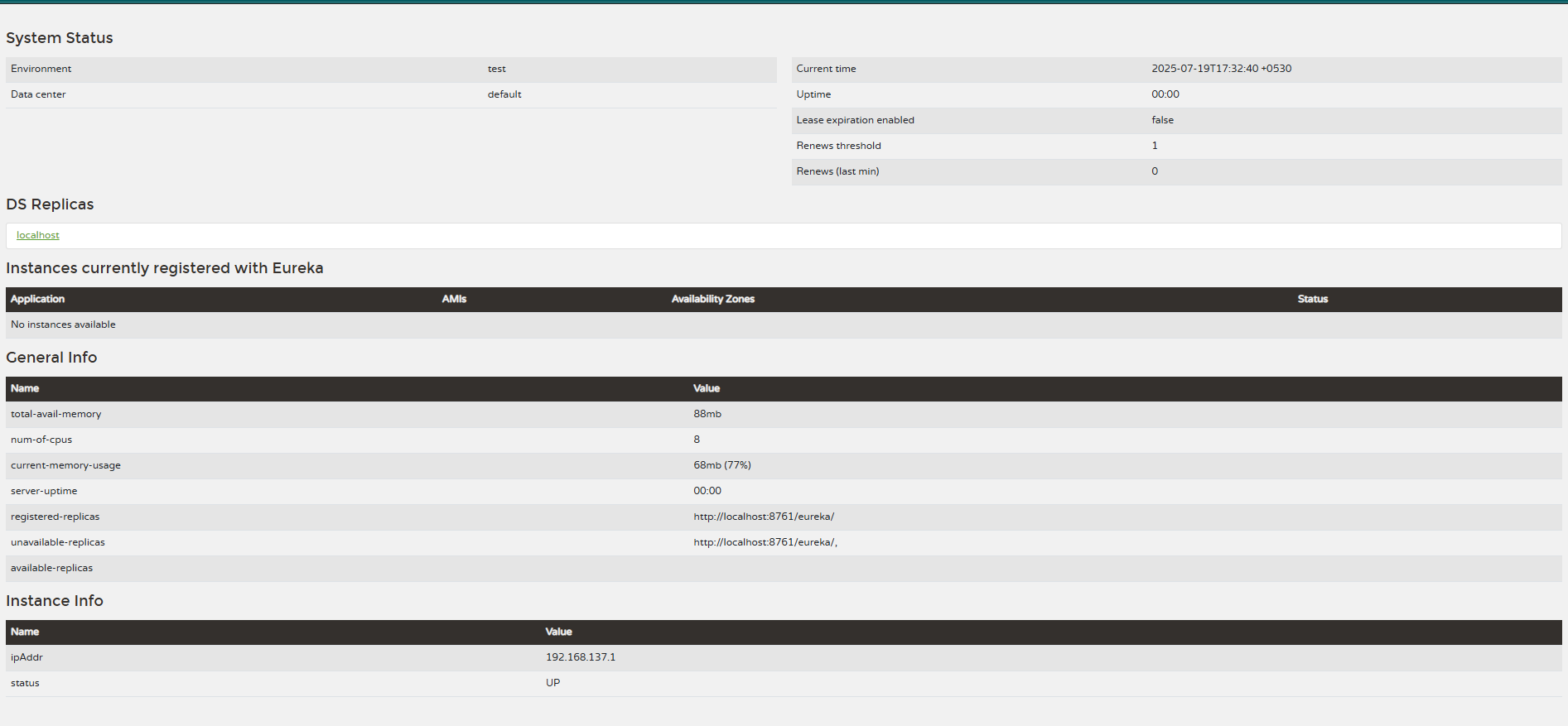
public static void main(String[] args) {

SpringApplication.*run*(LoanApplication.class, args);

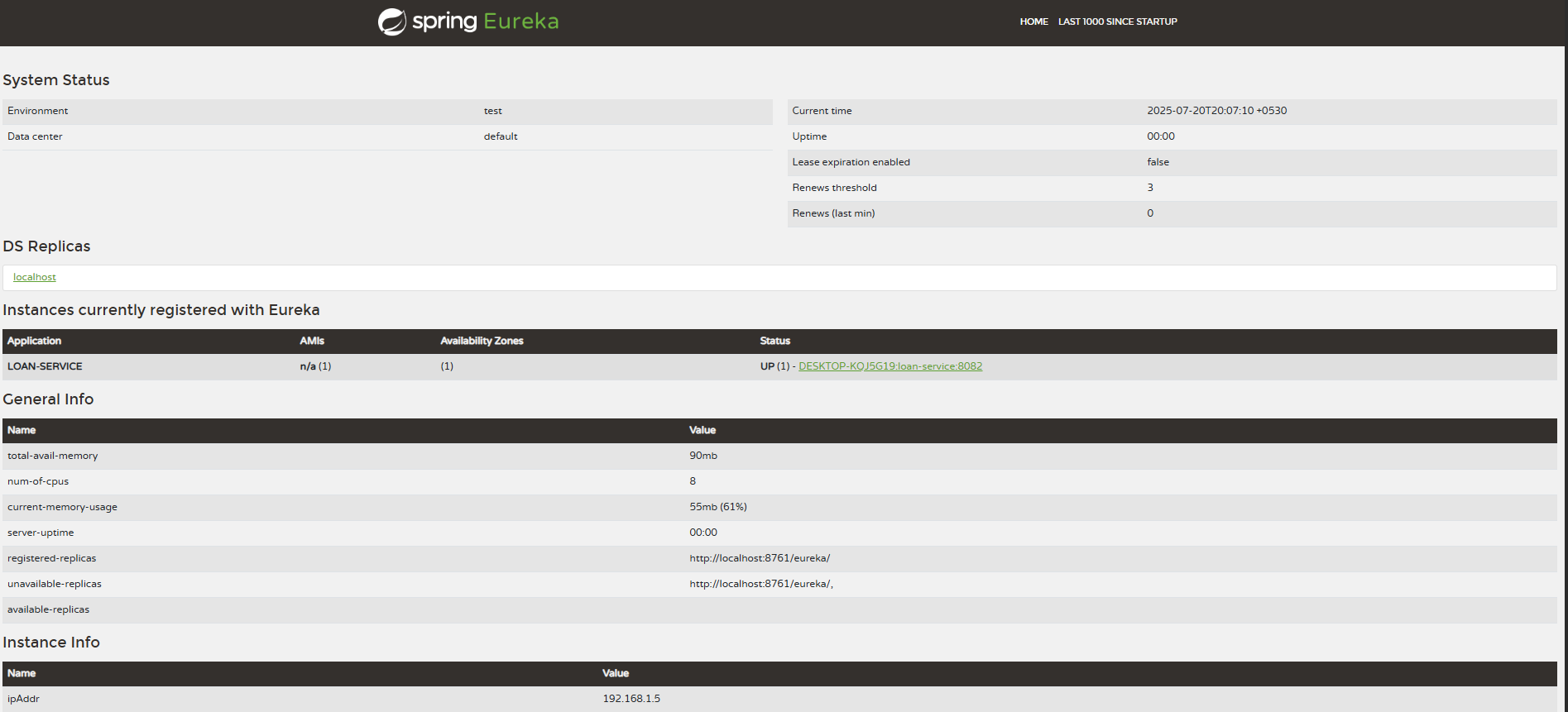
}

}

Upon runing eureka-discover-server



Secondly, upon running loan-service



1. For greet-service

package com.cognizant.greet\_service;

import org.springframework.web.bind.annotation.\*;

import java.util.\*;

*@RestController*

public class GreetController {

*@GetMapping*("/greet")

public String greet() {

return "Hello World";

}

}

package com.cognizant.greet\_service;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.cloud.client.discovery.EnableDiscoveryClient;

*@SpringBootApplication*

*@EnableDiscoveryClient*

public class GreetServiceApplication {

public static void main(String[] args) {

SpringApplication.*run*(GreetServiceApplication.class, args);

}

}

Console:

. \_\_\_\_ \_ \_\_ \_ \_

/\\ / \_\_\_'\_ \_\_ \_ \_(\_)\_ \_\_ \_\_ \_ \ \ \ \

( ( )\\_\_\_ | '\_ | '\_| | '\_ \/ \_` | \ \ \ \

\\/ \_\_\_)| |\_)| | | | | || (\_| | ) ) ) )

' |\_\_\_\_| .\_\_|\_| |\_|\_| |\_\\_\_, | / / / /

=========|\_|==============|\_\_\_/=/\_/\_/\_/

:: Spring Boot :: (v3.5.3)

2025-07-20T20:15:12.275+05:30 INFO 33236 --- [greet-service] [ restartedMain] c.c.g.GreetServiceApplication : Starting GreetServiceApplication using Java 21.0.7 with PID 33236 (D:\7282\microservices\greet-service\target\classes started by admin in D:\7282\microservices\greet-service)

2025-07-20T20:15:12.281+05:30 INFO 33236 --- [greet-service] [ restartedMain] c.c.g.GreetServiceApplication : No active profile set, falling back to 1 default profile: "default"

2025-07-20T20:15:12.394+05:30 INFO 33236 --- [greet-service] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : Devtools property defaults active! Set 'spring.devtools.add-properties' to 'false' to disable

2025-07-20T20:15:12.394+05:30 INFO 33236 --- [greet-service] [ restartedMain] .e.DevToolsPropertyDefaultsPostProcessor : For additional web related logging consider setting the 'logging.level.web' property to 'DEBUG'

2025-07-20T20:15:13.920+05:30 INFO 33236 --- [greet-service] [ restartedMain] o.s.cloud.context.scope.GenericScope : BeanFactory id=9536fd0b-8d58-3268-a2fb-285f9fa8febe

2025-07-20T20:15:14.611+05:30 INFO 33236 --- [greet-service] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8083 (http)

2025-07-20T20:15:14.641+05:30 INFO 33236 --- [greet-service] [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]

2025-07-20T20:15:14.641+05:30 INFO 33236 --- [greet-service] [ restartedMain] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]

2025-07-20T20:15:14.718+05:30 INFO 33236 --- [greet-service] [ restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext

2025-07-20T20:15:14.720+05:30 INFO 33236 --- [greet-service] [ restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 2324 ms

2025-07-20T20:15:15.820+05:30 INFO 33236 --- [greet-service] [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running on port 35729

2025-07-20T20:15:15.955+05:30 INFO 33236 --- [greet-service] [ restartedMain] DiscoveryClientOptionalArgsConfiguration : Eureka HTTP Client uses RestTemplate.

2025-07-20T20:15:16.035+05:30 WARN 33236 --- [greet-service] [ restartedMain] iguration$LoadBalancerCaffeineWarnLogger : Spring Cloud LoadBalancer is currently working with the default cache. While this cache implementation is useful for development and tests, it's recommended to use Caffeine cache in production.You can switch to using Caffeine cache, by adding it and org.springframework.cache.caffeine.CaffeineCacheManager to the classpath.

2025-07-20T20:15:16.083+05:30 INFO 33236 --- [greet-service] [ restartedMain] o.s.c.n.eureka.InstanceInfoFactory : Setting initial instance status as: STARTING

2025-07-20T20:15:16.135+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Initializing Eureka in region us-east-1

2025-07-20T20:15:16.144+05:30 INFO 33236 --- [greet-service] [ restartedMain] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints via configuration

2025-07-20T20:15:16.176+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Disable delta property : false

2025-07-20T20:15:16.177+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Single vip registry refresh property : null

2025-07-20T20:15:16.177+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Force full registry fetch : false

2025-07-20T20:15:16.177+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Application is null : false

2025-07-20T20:15:16.177+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Registered Applications size is zero : true

2025-07-20T20:15:16.177+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Application version is -1: true

2025-07-20T20:15:16.178+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Getting all instance registry info from the eureka server

2025-07-20T20:15:16.913+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : The response status is 200

2025-07-20T20:15:16.917+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Starting heartbeat executor: renew interval is: 30

2025-07-20T20:15:16.926+05:30 INFO 33236 --- [greet-service] [ restartedMain] c.n.discovery.InstanceInfoReplicator : InstanceInfoReplicator onDemand update allowed rate per min is 4

2025-07-20T20:15:16.929+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Discovery Client initialized at timestamp 1753022716928 with initial instances count: 1

2025-07-20T20:15:16.934+05:30 INFO 33236 --- [greet-service] [ restartedMain] o.s.c.n.e.s.EurekaServiceRegistry : Registering application GREET-SERVICE with eureka with status UP

2025-07-20T20:15:16.935+05:30 INFO 33236 --- [greet-service] [ restartedMain] com.netflix.discovery.DiscoveryClient : Saw local status change event StatusChangeEvent [timestamp=1753022716935, current=UP, previous=STARTING]

2025-07-20T20:15:16.938+05:30 INFO 33236 --- [greet-service] [foReplicator-%d] com.netflix.discovery.DiscoveryClient : DiscoveryClient\_GREET-SERVICE/DESKTOP-KQJ5G19:greet-service:8083: registering service...

2025-07-20T20:15:16.993+05:30 INFO 33236 --- [greet-service] [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8083 (http) with context path '/'

2025-07-20T20:15:16.995+05:30 INFO 33236 --- [greet-service] [ restartedMain] .s.c.n.e.s.EurekaAutoServiceRegistration : Updating port to 8083

2025-07-20T20:15:17.019+05:30 INFO 33236 --- [greet-service] [ restartedMain] c.c.g.GreetServiceApplication : Started GreetServiceApplication in 5.498 seconds (process running for 6.682)

2025-07-20T20:15:17.040+05:30 INFO 33236 --- [greet-service] [foReplicator-%d] com.netflix.discovery.DiscoveryClient : DiscoveryClient\_GREET-SERVICE/DESKTOP-KQJ5G19:greet-service:8083 - registration status: 204

1. For api-gateway

package com.cognizant.api\_gateway;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class ApiGatewayApplication {

public static void main(String[] args) {

SpringApplication.*run*(ApiGatewayApplication.class, args);

}

}

package com.cognizant.api\_gateway;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.cloud.gateway.filter.GlobalFilter;

import org.springframework.cloud.gateway.filter.GatewayFilterChain;

import org.springframework.core.Ordered;

import org.springframework.core.annotation.Order;

import org.springframework.stereotype.Component;

import org.springframework.web.server.ServerWebExchange;

import reactor.core.publisher.Mono;

*@Component*

public class LogFilter implements GlobalFilter, Ordered {

private static final Logger ***logger*** = LoggerFactory.*getLogger*(LogFilter.class);

*@Override*

public Mono<Void> filter(ServerWebExchange exchange, GatewayFilterChain chain) {

***logger***.info("Request Path: {}", exchange.getRequest().getPath());

return chain.filter(exchange);

}

*@Override*

public int getOrder() {

return -1;

}

}

Console and Output:

. \_\_\_\_ \_ \_\_ \_ \_

/\\ / \_\_\_'\_ \_\_ \_ \_(\_)\_ \_\_ \_\_ \_ \ \ \ \

( ( )\\_\_\_ | '\_ | '\_| | '\_ \/ \_` | \ \ \ \

\\/ \_\_\_)| |\_)| | | | | || (\_| | ) ) ) )

' |\_\_\_\_| .\_\_|\_| |\_|\_| |\_\\_\_, | / / / /

=========|\_|==============|\_\_\_/=/\_/\_/\_/

:: Spring Boot :: (v3.5.3)

2025-07-20T20:37:07.581+05:30 INFO 18480 --- [api-gateway] [ main] c.c.api\_gateway.ApiGatewayApplication : Starting ApiGatewayApplication using Java 21.0.7 with PID 18480 (D:\7282\microservices\api-gateway\target\classes started by admin in D:\7282\microservices\api-gateway)

2025-07-20T20:37:07.586+05:30 INFO 18480 --- [api-gateway] [ main] c.c.api\_gateway.ApiGatewayApplication : No active profile set, falling back to 1 default profile: "default"

2025-07-20T20:37:09.531+05:30 INFO 18480 --- [api-gateway] [ main] o.s.cloud.context.scope.GenericScope : BeanFactory id=8b518e1f-496e-339a-aef0-70114078920d

2025-07-20T20:37:09.654+05:30 WARN 18480 --- [api-gateway] [ main] trationDelegate$BeanPostProcessorChecker : Bean 'org.springframework.cloud.client.loadbalancer.reactive.LoadBalancerBeanPostProcessorAutoConfiguration' of type [org.springframework.cloud.client.loadbalancer.reactive.LoadBalancerBeanPostProcessorAutoConfiguration] is not eligible for getting processed by all BeanPostProcessors (for example: not eligible for auto-proxying). The currently created BeanPostProcessor [loadBalancerWebClientBuilderBeanPostProcessor] is declared through a non-static factory method on that class; consider declaring it as static instead.

2025-07-20T20:37:09.656+05:30 WARN 18480 --- [api-gateway] [ main] trationDelegate$BeanPostProcessorChecker : Bean 'org.springframework.cloud.client.loadbalancer.reactive.LoadBalancerBeanPostProcessorAutoConfiguration$ReactorDeferringLoadBalancerFilterConfig' of type [org.springframework.cloud.client.loadbalancer.reactive.LoadBalancerBeanPostProcessorAutoConfiguration$ReactorDeferringLoadBalancerFilterConfig] is not eligible for getting processed by all BeanPostProcessors (for example: not eligible for auto-proxying). Is this bean getting eagerly injected/applied to a currently created BeanPostProcessor [loadBalancerWebClientBuilderBeanPostProcessor]? Check the corresponding BeanPostProcessor declaration and its dependencies/advisors. If this bean does not have to be post-processed, declare it with ROLE\_INFRASTRUCTURE.

2025-07-20T20:37:09.660+05:30 WARN 18480 --- [api-gateway] [ main] trationDelegate$BeanPostProcessorChecker : Bean 'reactorDeferringLoadBalancerExchangeFilterFunction' of type [org.springframework.cloud.client.loadbalancer.reactive.DeferringLoadBalancerExchangeFilterFunction] is not eligible for getting processed by all BeanPostProcessors (for example: not eligible for auto-proxying). Is this bean getting eagerly injected/applied to a currently created BeanPostProcessor [loadBalancerWebClientBuilderBeanPostProcessor]? Check the corresponding BeanPostProcessor declaration and its dependencies/advisors. If this bean does not have to be post-processed, declare it with ROLE\_INFRASTRUCTURE.

2025-07-20T20:37:11.237+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [After]

2025-07-20T20:37:11.238+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [Before]

2025-07-20T20:37:11.238+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [Between]

2025-07-20T20:37:11.238+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [Cookie]

2025-07-20T20:37:11.238+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [Header]

2025-07-20T20:37:11.238+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [Host]

2025-07-20T20:37:11.238+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [Method]

2025-07-20T20:37:11.238+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [Path]

2025-07-20T20:37:11.238+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [Query]

2025-07-20T20:37:11.239+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [ReadBody]

2025-07-20T20:37:11.239+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [RemoteAddr]

2025-07-20T20:37:11.239+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [XForwardedRemoteAddr]

2025-07-20T20:37:11.239+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [Weight]

2025-07-20T20:37:11.239+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.g.r.RouteDefinitionRouteLocator : Loaded RoutePredicateFactory [CloudFoundryRouteService]

2025-07-20T20:37:11.786+05:30 INFO 18480 --- [api-gateway] [ main] DiscoveryClientOptionalArgsConfiguration : Eureka HTTP Client uses RestTemplate.

2025-07-20T20:37:11.874+05:30 WARN 18480 --- [api-gateway] [ main] iguration$LoadBalancerCaffeineWarnLogger : Spring Cloud LoadBalancer is currently working with the default cache. While this cache implementation is useful for development and tests, it's recommended to use Caffeine cache in production.You can switch to using Caffeine cache, by adding it and org.springframework.cache.caffeine.CaffeineCacheManager to the classpath.

2025-07-20T20:37:11.935+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.n.eureka.InstanceInfoFactory : Setting initial instance status as: STARTING

2025-07-20T20:37:12.215+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Initializing Eureka in region us-east-1

2025-07-20T20:37:12.225+05:30 INFO 18480 --- [api-gateway] [ main] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints via configuration

2025-07-20T20:37:12.289+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Disable delta property : false

2025-07-20T20:37:12.290+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Single vip registry refresh property : null

2025-07-20T20:37:12.290+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Force full registry fetch : false

2025-07-20T20:37:12.290+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Application is null : false

2025-07-20T20:37:12.290+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Registered Applications size is zero : true

2025-07-20T20:37:12.291+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Application version is -1: true

2025-07-20T20:37:12.291+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Getting all instance registry info from the eureka server

2025-07-20T20:37:13.186+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : The response status is 200

2025-07-20T20:37:13.191+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Starting heartbeat executor: renew interval is: 30

2025-07-20T20:37:13.196+05:30 INFO 18480 --- [api-gateway] [ main] c.n.discovery.InstanceInfoReplicator : InstanceInfoReplicator onDemand update allowed rate per min is 4

2025-07-20T20:37:13.209+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Discovery Client initialized at timestamp 1753024033207 with initial instances count: 1

2025-07-20T20:37:13.213+05:30 INFO 18480 --- [api-gateway] [ main] o.s.c.n.e.s.EurekaServiceRegistry : Registering application API-GATEWAY with eureka with status UP

2025-07-20T20:37:13.215+05:30 INFO 18480 --- [api-gateway] [ main] com.netflix.discovery.DiscoveryClient : Saw local status change event StatusChangeEvent [timestamp=1753024033215, current=UP, previous=STARTING]

2025-07-20T20:37:13.218+05:30 INFO 18480 --- [api-gateway] [nfoReplicator-0] com.netflix.discovery.DiscoveryClient : DiscoveryClient\_API-GATEWAY/DESKTOP-KQJ5G19:api-gateway:9090: registering service...

2025-07-20T20:37:13.336+05:30 INFO 18480 --- [api-gateway] [nfoReplicator-0] com.netflix.discovery.DiscoveryClient : DiscoveryClient\_API-GATEWAY/DESKTOP-KQJ5G19:api-gateway:9090 - registration status: 204

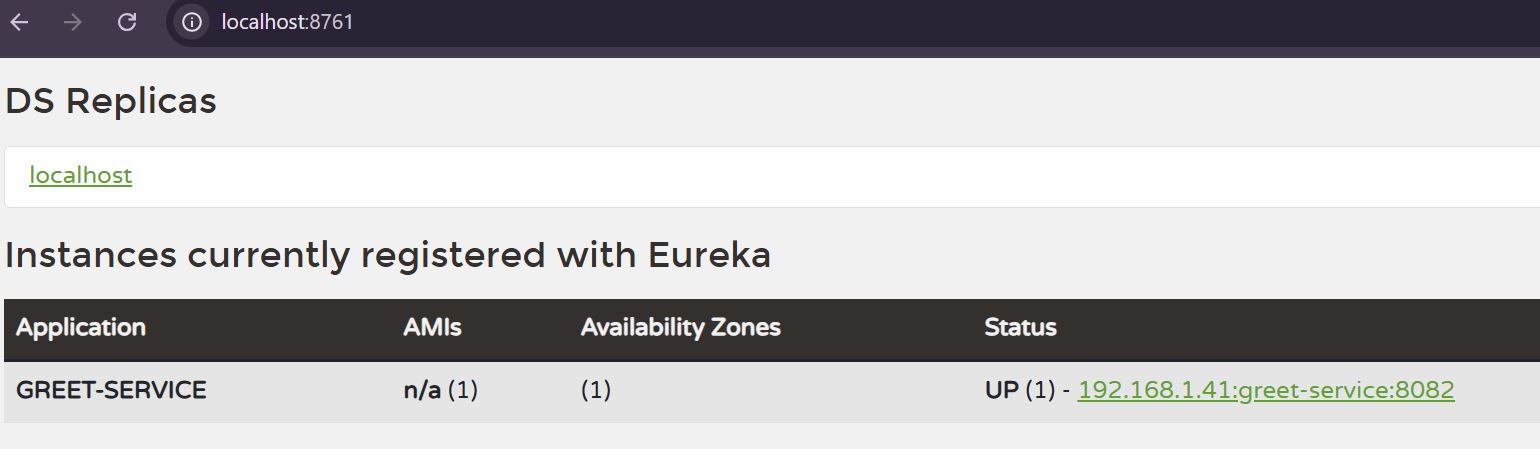
2025-07-20T20:37:13.649+05:30 INFO 18480 --- [api-gateway] [ main] o.s.b.web.embedded.netty.NettyWebServer : Netty started on port 9090 (http)

2025-07-20T20:37:13.652+05:30 INFO 18480 --- [api-gateway] [ main] .s.c.n.e.s.EurekaAutoServiceRegistration : Updating port to 9090

2025-07-20T20:37:13.686+05:30 INFO 18480 --- [api-gateway] [ main] c.c.api\_gateway.ApiGatewayApplication : Started ApiGatewayApplication in 6.926 seconds (process running for 7.655)

2025-07-20T20:37:55.520+05:30 INFO 18480 --- [api-gateway] [ctor-http-nio-2] com.cognizant.api\_gateway.LogFilter : Request Path: /greet-service/greet

2025-07-20T22:14:15.350+05:30 INFO 18480 --- [api-gateway] [trap-executor-0] c.n.d.s.r.aws.ConfigClusterResolver : Resolving eureka endpoints via configuration





**Microservice REST API Outputs**

**Account Service**

<http://localhost:8080/accounts/123456>

{

"number": "123456",

"type": "savings",

"balance": 234343

}

**Loan Service**

<http://localhost:8081/loans/H98765>

{

"number": "H98765",

"type": "car",

"loan": 400000,

"emi": 3258,

"tenure": 18

}